

**REMARKS**

Claims 1, 2, 4-20 and 22-34 are pending in the present application. Claim 21 was canceled; claims 1, 9, 13, 18, 19, 27, 31 and 32 were amended; claims 33 and 34 were added. Reconsideration of the claims is respectfully requested.

**I. 35 U.S.C. § 101, Double Patenting**

The Examiner has objected to claim 21 under 37 CFR 1.75 as being a substantial duplicate of claim 20. In response, Applicants have cancelled claim 21, and the double patenting rejection is now believed moot.

**II. 35 U.S.C. § 102, Anticipation, Claims 1-2, 4-5, 9-10, 13-23, 27-28, and 31-32**

The Examiner has rejected claims 1-2, 4-5, 9-10, 13-23, 27-28, and 31-32 under 35 U.S.C. § 102 as being anticipated by Brighton Beach Software (referred as Brighton Beach), CodeSpell: Source Code Spell Checker [BETA], <http://www.bbs.com.au/codespel.htm>, November 1998, Google, printouts pages 1-2. This rejection is respectfully traversed.

In rejecting claims 1-2, 4-5, 9-10, 13-23, 27-28, and 31-32, the Office Actions states:

Regarding independent claim 1 and its dependent claim 4, Brighton Beach discloses:

receiving computer source code for processing (page 1, Description: the fact that spell checking the programs of the programmers inherently shows that the computer source code in the programs is received for processing)

identifying displayable text within the computer source code where the text located between a set of delimiters is considered as the displayable text (page 1, Description: the fact that the text strings within the programs of the programmers can be extracted, saved in files and reviewed for logical correctness and consistency where the files are stored in ASCII format with *string delimiters such as single quotes or double-quotes* shows that the text strings between the string delimiters in the source code are identified as displayable text)

checking the displayable text for errors (page 1, Description: spell check the displayable text strings in the programs)

(Office Action dated September 10, 2004, pages 3 and 4).

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). The Brighton Beach reference cited by the Examiner does not anticipate the present invention as recited in claim 1, because Brighton Beach fails to teach each and every element of claim 1.

Independent claim 1, which is representative of independent claims 13, 19 and 31 with regard to similarly recited subject matter, recites:

1. A method in a data processing system for spell checking text, the method comprising:
  - receiving computer source code for processing;
  - responsive to a determination of a source code format, identifying displayable text within the computer source code; and
  - checking the displayable text for errors.

Brighton Beach does not teach each and every feature of the presently claimed invention in claim 1. Brighton Beach does not teach the feature of "responsive to a determination of a source code format, identifying displayable text within the computer source code." Brighton Beach states, in the description on page 1 "CodeSpell will work with any programming language that has files stored in ASCII format and uses delimiters such as "or '." Therefore, Brighton Beach only works on ASCII files. Thus, Brighton Beach does not teach making or have any reason to make a determination of the source code format, as it is designed only for files stored in an ASCII format. In contrast, the present invention provides advantages and capabilities not found in the prior art, as the present invention is not limited to using source code in particular format. Therefore, Brighton Beach does not teach the feature of "responsive to a determination of a source code format, identifying displayable text within the computer source code." Thus, Brighton Beach does not anticipate the present invention as recited in claim 1, because Brighton Beach fails to teach each and every element of claim 1.

Additionally, Brighton Beach can only check strings which have single or double quotes as string delimiters, as stated on page 1 of Brighton Beach and by The Windows Developer Network information regarding Brighton Beach provided by the Examiner,

wysiwyg://1/http://www.windevnet.com/lonk/subject166.htm?topic= links. In contrast, the present invention provides advantages and capabilities not found in the prior art, as the present invention can identify any displayable text, no matter how the displayable text is denoted within the source code. Therefore, Brighton Beach does not teach the feature of "responsive to a determination of a source code format, identifying displayable text within the computer source code." Thus, Brighton Beach does not anticipate the present invention as recited in claim 1, because Brighton Beach fails to teach each and every element of claim 1.

Furthermore, the ASCII format has several limitations, which the present invention overcomes. For instance, while ASCII works for the English language, it does not work for many eastern languages, such as Chinese. That is, ASCII cannot be used to express text in these languages. The reason for this is that ASCII uses a single byte for each character. However, many languages, such as Japanese and Chinese, require the use of 2 bytes for a single character. Therefore, the Brighton Beach invention would not be able to recognize text strings in foreign languages as it would try to interpret each byte separately. In contrast, the present invention can detect and recognize displayable text of various languages within the source code. Further the present invention is not limited to using source code in a particular format. Brighton Beach teaches only checking text delimited by either single or double quotes and only in the ASCII language. Thus, Brighton Beach does not anticipate the present invention as recited in claim 1, because Brighton Beach fails to teach each and every element of claim 1.

Therefore, for all the reasons stated above, Applicants believe that Brighton Beach does not teach all the features of independent claims 1, 13, 19 and 31. Accordingly, Applicants respectfully submit that claims 1, 13, 19 and 31 are patentable over the Brighton Beach reference.

Regarding independent claim 9, the Office Action states:

As per independent claim 9, Brighton Beach discloses:  
searching source code for a first delimiter indicative of displayable text (page 1, Description: the fact that Code.Spell works with any programming language that has files stored in ASCII format and uses delimiters such as single quotes or double quotes inherently shows that to spell check the text strings in the programs, which are equivalent to the

source code, Code.Spell has to search the source code in the programs for the displayable text between the two delimiters since delimiters need no spell checking; in other words, this searching includes searching for the first delimiter indicative of displayable text)

responsive to finding the first, spell checking text after the first delimiter until a second delimiter is encountered (page 1, Description: as mentioned above, Code.Spell checks spells the text between two delimiters; in other words, spell checking text after the first delimiter until a second delimiter is performed)

(Office Action dated September 10, 2004, page 5).

The Brighton Beach reference cited by the Examiner does not anticipate the present invention as recited in claim 9, because Brighton Beach fails to teach each and every element of claim 9. Independent claim 9, which is representative of independent claims 18, 27 and 32 with regard to similarly recited subject matter, recites:

9. A method in a data processing system for checking text, the method comprising:  
responsive to a determination of a source code format, searching the source code for a first delimiter indicative of displayable text; and  
responsive to finding the first delimiter, spell checking text after the first delimiter until a second delimiter is encountered.

Brighton Beach does not teach each and every feature of the presently claimed invention in claim 9. Brighton Beach does not teach the feature of "responsive to a determination of a source code format, searching source code for a first delimiter indicative of displayable text." Brighton Beach states, in the description on page 1 "Code.Spell will work with any programming language that has files stored in ASCII format and uses delimiters such as " or '." Therefore, Brighton Beach only works on ASCII files that have a single or double quote as a string delimiter. Thus, Brighton Beach does not teach making a determination of a source code format, as it only works for files stored in an ASCII format. In contrast, the present invention provides advantages and capabilities not found in the prior art, as the present invention can detect and recognize displayable text of various languages within the source code, no matter how the displayable text is denoted within the source code. Further the present invention is not limited to using source code in a particular format. Therefore, Brighton Beach does not teach the feature of "responsive to a

determination of a source code format, searching source code for a first delimiter indicative of displayable text.” Thus, Brighton Beach does not anticipate the present invention as recited in claim 9, because Brighton Beach fails to teach each and every element of claim 9.

Therefore, for all the reasons stated above, Applicants believe that Brighton Beach does not teach all the features of independent claims 9, 18, 27 and 32. Accordingly, Applicants respectfully submit that claims 9, 18, 27 and 32 are patentable over the Brighton Beach reference.

Claims 2, 4, 5 and 33 are dependent claims depending from independent claim 1. Claims 10 and 34 are dependent claims depending from independent claim 9. Claims 14-17 are dependent claims depending from independent claim 13. Claims 20, 22 and 23 are dependent claims depending from independent claim 19. Claims 28 and 29 are dependent claims depending from independent claim 27. As Applicants have already demonstrated that claims 1, 9, 13, 19 and 27 are patentable over the Brighton Beach reference, Applicants submit that dependent claims 2, 4, 5, 10, 14-17, 20, 22, 23, 28, 29, 33 and 34 are also patentable over the Brighton Beach reference, at least by virtue of depending from an allowable claim. Consequently, Applicants respectfully submit that the rejection of claims 2, 4, 5, 10, 14-17, 20, 22, 23, 28 and 29 has been overcome. Additionally, several claims recite other additional combinations of features not suggested by the reference.

For example, new claim 33 recites the feature of “wherein the set of delimiters are custom delimiters determined by an author of the source code.” New claim 34 recites the similar feature of “wherein the first and second delimiters are custom delimiters determined by an author of the source code.” These features are not taught by Brighton Beach. Brighton Beach teaches in the description on page 1 “Code.Spell will work with any programming language that has files stored in ASCII format and uses delimiters such as “ or ‘.” Therefore, Brighton Beach only works on ASCII files that have a single or double quote as a string delimiter. Thus, Brighton Beach does not teach the ability to recognize custom delimiters, determined by the programmer, used to indicate strings of text. Therefore Brighton Beach does not teach the feature of either “wherein the set of delimiters are custom delimiters determined by an author of the source code” or “wherein

the first and second delimiters are custom delimiters determined by an author of the source code." Thus, Brighton Beach does not anticipate the present invention as recited in claims 33 and 34, because Brighton Beach fails to teach each and every element of claims 33 and 34.

Therefore, the rejection of claims 1-2, 4-5, 9-10, 13-23, 27-28, and 31-32 under 35 U.S.C. § 102 has been overcome.

### III. 35 U.S.C. § 103, Obviousness, Claims 6-7, 11, 24-25, and 29

The Examiner has rejected claims 6-7, 11, 24-25, and 29 under 35 U.S.C. § 103 as being unpatentable over Brighton Beach as applied to claims 1 and 9 above, and further in view of Kawanabe (US Pat No. 5,924,059, 7/13/99, filed 2/17/94). This rejection is respectfully traversed.

In rejecting claims 6-7, 11, 24-25, and 29, the Office Actions states:

Regarding claim 6, which is dependent on claim 1, Brighton Beach does not disclose explicitly that checking step includes:  
selecting a dictionary  
spell checking the displayable text using the dictionary  
Kawanabe discloses spell checking includes:  
selecting a dictionary (col 1, lines 53-58, figure 7: detecting the presence/absence of a dictionary memory used for checking word spelling in dictionary ROM inherently shows that the dictionary is selected in case when the dictionary is present for checking spelling)  
spell checking the displayable text using the dictionary (abstract, figure 6, col 1, lines 49-63)

It would have been obvious to one of ordinary skill in the art at the time of the invention was made to have combined Kawanabe into Brighton Beach since Kawanabe teaches using a dictionary for spell checking text in a document providing the advantage to incorporate into Brighton Beach for using a dictionary for spell checking the displayable text in the source code program since said text is the same as any text in a typical document.

Regarding claim 7, which is dependent on claim 1, Brighton Beach does not disclose that the dictionary is selected using a user input. Kawanabe discloses the dictionary is selected using a user input (figure 4, col 3, lines 32-34: since the routine (detecting the presence/absence of the dictionary for selecting the dictionary) is started in response to a reset signal upon power-on of the typewriter, which is considered as an input from a user).

It would have been obvious to one of ordinary skill in the art at the time of the invention was made to have combined Kawanabe into Brighton Beach since Kawanabe discloses selecting a dictionary upon a user input providing the advantage to incorporate into Brighton Beach for offering a user a selection of a dictionary.

(Office Action dated September 10, 2004, pages 7-8).

A fundamental notion of patent law is the concept that invention lies in the new combination of old elements. Therefore, a rule that every invention could be rejected as obvious by merely locating each element of the invention in the prior art and combining the references to formulate an obviousness rejection is inconsistent with the very nature of "invention." Consequently, a rule exists that a combination of references made to establish a prima facie case of obviousness must be supported by some teaching, suggestion, or incentive contained in the prior art which would have led one of ordinary skill in the art to make the claimed invention.

The Examiner bears the burden of establishing a prima facie case of obviousness based on the prior art when rejecting claims under 35 U.S.C. § 103. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780 (Fed. Cir. 1992).

Claims 6 and 7 are dependent claims depending from claim 1. Claim 11 is a dependent claim depending from claim 9. Claims 24 and 25 are dependent claims depending from claim 19. Claim 29 is dependent claim depending from claim 27. The Brighton Beach reference still does not teach or suggest all the features of claims 1, 9, 19 and 27, as argued above.

Furthermore, Kawanabe does not cure the deficiencies of Brighton Beach. Kawanabe does not teach the features missing from Brighton Beach, including "responsive to a determination of a source code format, identifying displayable text within the computer source code" and "responsive to a determination of a source code format, searching source code for a first delimiter indicative of displayable text," nor does the Examiner point to any portion of Kawanabe that teaches these features.

Additionally, when combined, Brighton Beach and Kawanabe do not teach the presently claimed invention as recited in claims 6 and 24. Specifically, claims 6 and 24 recite the feature of "selecting a dictionary." The Examiner admits that Brighton Beach

does not teach this feature and points to column 1, lines 53-58 in Kawanabe as teaching this feature:

It is another object of the present invention to provide a document processing method and apparatus, which comprises control means for detecting the presence/absence of a dictionary memory used for checking word spelling, and for enabling or disabling the spell checker function on the basis of the detection result.

The above cited passage does not teach "selecting a dictionary." The passage teaches detecting whether or not dictionary memory is present. The passage does not teach selecting a dictionary. Nowhere does Kawanabe teach selecting a dictionary. All that Kawanabe teaches is detecting if dictionary memory is present and then enabling/disabling a spell checking function based on this determination. The mere fact that a typewriter has dictionary memory does not mean that a dictionary is selected. Selecting means choosing from a number of alternatives, if there is only one dictionary, there is no selecting involved. Therefore, the proposed combination of the references does not teach the presently claimed invention as recited in claims 6 and 24. Thus, the Examiner has failed to state a prima facie case of obviousness.

Regarding claims 7 and 25, Brighton Beach and Kawanabe do not teach the presently claimed invention as recited in claims 7 and 15. Specifically, claims 7 and 25 recite the feature of "wherein the dictionary is selected using a user input." The Examiner admits that Brighton Beach does not teach this feature and points to column 3, lines 32-34 in Kawanabe as teaching this feature:

FIG. 4 shows the main routine of the control operation of the CPU  
1. This routine is started in response to a reset signal upon power-on of the typewriter.

The above cited passage does not teach the feature of "wherein the dictionary is selected using a user input." Instead, the above cited passage teaches that the process of detecting whether dictionary memory is present begins when the typewriter is turned on. As was discussed above in regards to claim 6, simply detecting whether or not something exists is not equivalent to selecting one item from among a group of alternatives. Therefore, the



proposed combination of the references does not teach the presently claimed invention as recited in claims 7 and 25. Thus, the Examiner has failed to state a prima facie case of obviousness.

Claims 11 and 29 recite the feature of "wherein the text is checked using a selected dictionary." The Examiner admits that Brighton Beach does not teach this feature and points to Figure 4 in Kawanabe as teaching this feature. However, Figure 4 does not teach this feature. As argued above in regards to claims 6, 7, 24 and 25, Kawanabe does not teach selecting a dictionary. Therefore it follows that Kawanabe does not teach "wherein the text is checked using a selected dictionary." Therefore the proposed combination of the references does not teach the presently claimed invention as recited in claims 11 and 29. Thus, the Examiner has failed to state a prima facie case of obviousness.

Thus claims 6, 7, 11, 24, 25 and 29 are patentable over the cited references since the proposed combination of the Brighton Beach reference with Kawanabe would not reach the presently claimed invention. The features relied upon as being taught in the Brighton Beach reference are not taught or suggested by that reference, as explained above. Kawanabe does not cure the deficiencies of Brighton Beach. As a result, a combination of these references would not reach the claimed invention in claims 6, 7, 11, 24, 25 and 29.

In view of the above, Applicants submit that dependent claims 6, 7, 11, 24, 25 and 29 are not taught or suggested by Brighton Beach or Kawanabe. Claims 6, 7, 11, 24, 25 and 29 are dependent claims depending on independent claims 1, 9, 19 and 27. Applicants have already demonstrated claims 1, 9, 19 and 27 to be condition for allowance. Applicants respectfully submit that claims 6, 7, 11, 24, 25 and 29 are also allowable, at least by virtue of their dependency on allowable claims as well as their including features not taught or suggested by either the Brighton Beach reference or the Kawanabe reference.

Therefore, the rejection of claims 6-7, 11, 24-25, and 29 under 35 U.S.C. § 103 has been overcome.

**IV. 35 U.S.C. § 103, Obviousness, Claims 8, 12, 26 and 30**

The Examiner has rejected claims 8, 12, 26 and 30 under 35 U.S.C. § 103 as being unpatentable over Brighton Beach. This rejection is respectfully traversed.

The Brighton Beach reference still does not teach or suggest all the features of claims 1, 9, 19 and 27, as argued above. Claim 8 is a dependent claim depending on claim 1, Claim 12 is a dependent claim depending on claim 9. Claim 26 is a dependent claim depending on claim 19. Claim 30 is a dependent claim depending on claim 27. As Applicants have already demonstrated claims 1, 9, 19 and 27 to be condition for allowance, Applicants respectfully submit that claims 8, 12, 26 and 30 are also allowable, at least by virtue of their dependency on allowable claims as well as their including features not taught or suggested by the Brighton Beach reference.

For example, claims 8 and 26 recite the feature of "locating a pointer in the source code to a resource file containing the displayable text." The Examiner admits that Brighton Beach does not teach this feature. The Examiner has stated that the addition of this feature would have been obvious to anyone skilled in the art as Brighton Beach includes the feature of extracting, saving or printing and reviewing the natural text strings for logical correctness and consistency. These functions of Brighton Beach do not teach or suggest the feature of "locating a pointer in the source code to a resource file containing the displayable text," as recited in claims 8 and 26 of the presently claimed invention. Nowhere does Brighton Beach disclose how these functions are preformed. There is nothing that states that Brighton Beach, when extracting the text strings, creates a pointer in the source code for each string, which refers to a resource file wherein the text strings are stored separately.

A proper prima facie case of obviousness must be supported by some teaching or suggestion contained in the prior art. Nowhere does Brighton Beach teach, suggest, or give any incentive for "locating a pointer in the source code to a resource file containing the displayable text." Brighton Beach only teaches a spell checker for use in checking text strings in source code stored in an ASCII format only, using only single and double quotes as strings delimiters. No suggestion of a combination of components necessary to "locating a pointer in the source code to a resource file containing the displayable text," is found in Brighton Beach. The Examiner has not pointed out any teaching, suggestion, or

incentive in the prior art to add the feature "locating a pointer in the source code to a resource file containing the displayable text." Therefore, the presently claimed invention can only be reached through an impermissible use of hindsight. One cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention. *In re Fritch*, 972 F.2d 1260, 23 U.S.P.Q.2d 1780, 1784 (Fed. Cir. 1992) Thus, the Examiner has failed to state a prima facie case of obviousness.

Thus, for all the reasons stated above, claims 8, 12, 26 and 30 are patentable over the cited reference since the features relied upon as being taught in the Brighton Beach reference are not taught or suggested by that reference, as explained above.

Therefore, the rejection of claims 8, 12, 26, and 30 under 35 U.S.C. § 103 has been overcome.

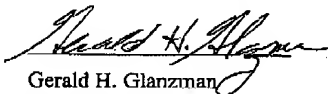
V. Conclusion

It is respectfully urged that the subject application is patentable over the cited references and is now in condition for allowance.

The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: December 10, 2004

Respectfully submitted,



Gerald H. Glanzman  
Reg. No. 25,035  
Yee & Associates, P.C.  
P.O. Box 802333  
Dallas, TX 75380  
(972) 385-8777  
Attorney for Applicants

AI/bj